Predicting Movie Box Office Revenue Using Machine Learning

1st Anant Sahoo
Department of EECS
University of Tennessee, Knoxville
Knoxville, United States
asahoo@vols.utk.edu

2nd Deep Patel
Department of EECS
University of Tennessee, Knoxville
Knoxville, United States
dpate125@vols.utk.edu

3rd Sulaiman Mohyuddin
Department of EECS
University of Tennessee, Knoxville
Knoxville, United States
smohyud1@vols.utk.edu

4th William Douglass
Department of EECS
University of Tennessee, Knoxville
Knoxville, United States
wdougla4@vols.utk.edu

5th William Sessoms
Department of EECS
University of Tennessee, Knoxville
Knoxville, United States
wsessoms@vols.utk.edu

***Abstract*—This project develops a machine learning model to predict box office revenue using only features available prior to a film's release. Using the TMDB 5000 dataset, we performed extensive data cleaning, JSON parsing, categorical encoding, and feature engineering to construct a structured representation of film characteristics. We engineered actor and director performance scores, applied multiple scaling techniques, and used a chronological train–test split to avoid data leakage. A Multi-Layer Perceptron Regressor (MLPRegressor) was trained and evaluated using Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the coefficient of determination (R^2). The final model achieved a validation MAE of $71.9M and an R^2 of 0.6361, demonstrating competitive performance given the noise and variability in box office outcomes. The results show strong predictive power for low- and mid-grossing films, with underestimation occurring primarily for breakout hits. This work highlights the challenges of modeling film performance and suggests opportunities for improvement through larger datasets, better representation of marketing effects, and the inclusion of non-English film markets.**

***Index Terms*—machine learning, regression, box office prediction, neural networks, MLPRegressor, feature engineering, film analytics, data preprocessing**

I. Introduction

A. Objective

The objective of this project is to build a regression model that predicts the box office revenue of movies. The model will use features available before a film's release, such as budget, genres, production company, language, keywords, and key cast members. By focusing on pre-release data, the model will be designed to estimate financial success ahead of time, allowing stakeholders to make informed decisions before production and release.

B. Motivation

Predicting box office revenue has practical value for multiple stakeholders in the film industry. Film production is a high-risk investment: a typical studio film can require tens or even hundreds of millions of dollars in funding before release, and marketing expenses often match or exceed the budget itself. Despite such high costs, the success of a movie is often unpredictable, making financial forecasting an important area for improvement. Being able to estimate the outcome of a project before release can reduce uncertainty for studios and investors. Accurate predictions can help determine whether a project is worth funding, guide marketing and promotion strategies, and even influence casting or directing decisions.

The model may also reveal broader trends in audience preferences across genres, budgets, and production strategies, creating opportunities for studios to align future projects with market demand. For example, studios may discover that certain combinations of genres, release seasons, or production companies consistently yield higher returns. Beyond its practical applications, this project provides a chance to apply machine learning techniques to a real-world dataset, offering both technical experience and industry insight. It also demonstrates how data-driven approaches can complement traditional decision-making in fields that are highly creative but financially constrained.

II. Data Description

A. Dataset Overview

For this project, we use the TMDB 5000 Dataset from Kaggle [1], which includes information on approximately 5,000 films. The dataset is split across two CSV files: one containing credits (title, cast, and crew) and another containing detailed movie statistics such as budget, revenue, popularity, production company, release date, runtime, genres, vote averages, and vote counts.

The credits file requires preprocessing because the cast and crew fields are stored as JSON strings. These are parsed into structured formats so we can extract key information, such as the director and the top-billed cast members. Monetary values

such as budget and revenue are adjusted for inflation to make earnings from different decades comparable.

While the dataset is fairly comprehensive, it does include challenges such as missing values and zero entries in budget or revenue fields. These issues will be addressed during cleaning and filtering. Beyond cleaning, we also plan to engineer several features. Categorical variables—such as genre, production company, and release month—can be encoded to capture meaningful patterns, and cast and crew information can be aggregated into summary statistics about their past success. Finally, we identify potential outliers, such as films with unusually high or low reported values, since these can distort model behavior. Preparing the dataset in a structured and reliable format ensures that the features used in the model accurately reflect real-world film characteristics.

B. Data Processing

The first step in our processing pipeline was to merge the two CSV files using the movie ID in order to form a unified dataset. After completing the merge, we removed columns that were not relevant to our analysis, including *homepage*, *overview*, *tagline*, *status*, *title*, and the movie ID once it was no longer required. Rows containing missing values were then removed to ensure that all remaining entries were usable for model training. Finally, we filtered the dataset to include only English-language films. English films represented the majority of usable samples, and non-English films may require separate modeling assumptions due to cultural and market differences.

Several fields in the dataset were stored as JSON strings, including *genres*, *keywords*, *production_companies*, *production_countries*, *spoken_languages*, and *cast*. These fields were parsed into structured formats. For the cast data, we extracted only the top five billed actors for each film to limit dimensionality. We also identified each film's director by parsing the *crew* field, which required additional handling for the small number of movies credited with multiple directors.

Next, we performed one-hot encoding on the genre information using a `MultiLabelBinarizer`, which generates a binary indicator column for each genre category. We also parsed release dates into additional temporal features, including the release year and release quarter, since timing can influence box office performance.

We engineered two additional features: an actor score and a director score. These scores summarize how "strong" the top five actors and the director are, based on their past work and performance. The idea is to quantify how much value talented actors or directors tend to bring to a movie. We accomplished this by taking the 5 most recent movies the actor or director had been in, and weighting their revenue with the largest weights on the most recent movies. We implemented this with a rolling window so that there is no data leakage in allowing the model to look ahead at future performances. Including these measures gives the model more context regarding the expected draw of the cast and crew, which in turn can improve the accuracy of box office predictions.

III. Methodology

To create our model, we used a standard Scikit-Learn `MLPRegressor` pipeline. We applied different scaling techniques based on the behavior of each feature. The `StandardScaler` was used for `budget`, `actor_score`, and `director_score` to highlight values that deviate from the mean. In contrast, we used a `MinMaxScaler` for `runtime`, `release_year`, and `release_quarter` to preserve smaller differences in those features.

To avoid data leakage, we split the training and testing sets chronologically. All films released in 2011 or earlier were assigned to the training set, and all films released after 2011 were assigned to the test set. This ensured that the model only used information available prior to the prediction period.

After scaling, the processed features were passed into the `MLPRegressor`. Because the model trained quickly, we were able to manually search for a set of effective hyperparameters. To evaluate performance, we examined the Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the coefficient of determination (R^2). These metrics allowed us to assess prediction accuracy as well as the amount of variability captured by the model.

The final model architecture consisted of three hidden layers with sizes 128, 64, and 32, along with a learning rate of 0.005.

IV. Results

To evaluate our model, we used two primary metrics: Mean Absolute Error (MAE) and the coefficient of determination (R^2). After tuning for the most effective hyperparameters, the model achieved a training MAE of \$51.4M and a validation MAE of \$71.9M. This indicates that, on average, the predictions on the validation set were approximately \$72M away from the true box office values. Although this may appear large, the movies in our dataset were released after 2011, and many of them involve budgets and revenues where a \$72M deviation represents a relatively small proportion of the overall scale.

The model also obtained an R^2 value of 0.6653 on the training set and 0.6361 on the validation set. Despite the difference in MAE, the closeness of these R^2 scores suggests that the model is not overfitting. Given the high variance in movie earnings and the unpredictability of the post-2011 era—including the COVID-19 pandemic and periods of rapid inflation—we consider these R^2 values to be reasonably strong for a domain with such noisy target variables.

Figures Fig. 1 and Fig. 2 present the model's performance on the validation set. As shown, the model performs well for low- and mid-grossing films but struggles with several breakout hits that strongly deviate from typical patterns. The plots also show that most of the larger errors occur when the model underestimates earnings. We believe this trend may be partially explained by inflation and other economic shifts that are not directly captured by the available features. Overall, the results indicate that the model produces stable predictions, avoids overfitting, and provides a solid foundation for future improvements.

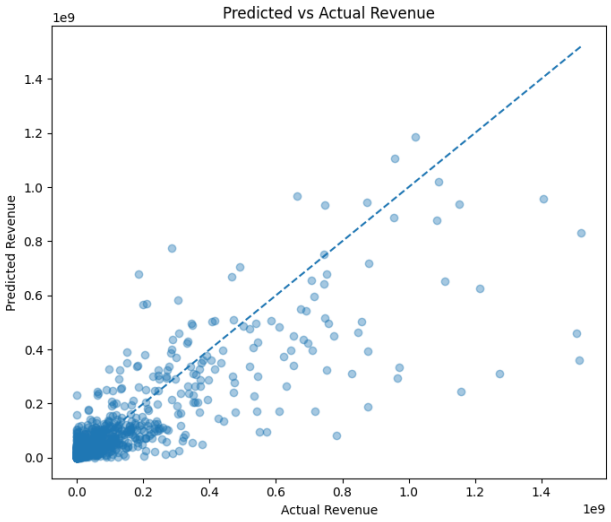

Fig. 1. Predicted Versus Actual Revenue.

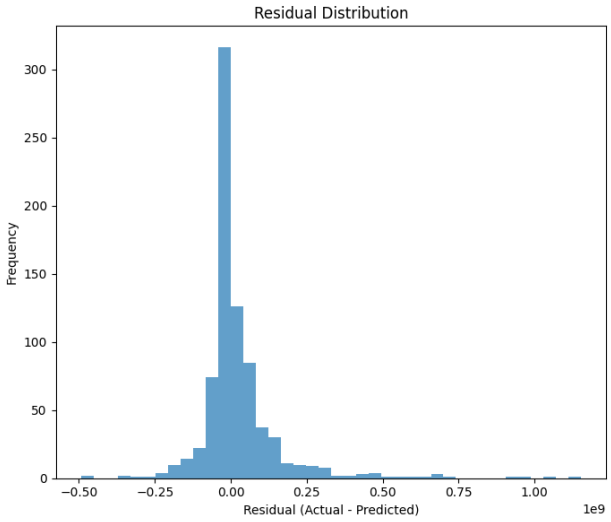

Fig. 2. Residual Distribution.

V. Challenges and Limitations

Many of the challenges we encountered were related to processing the JSON objects that make up the cast and crew columns. This was especially difficult for the *crew* field, which includes a wide range of roles that do not necessarily contribute to a film's box office value. Positions such as makeup artists, costume designers, and various technical staff, while essential to production, do not have a measurable impact on consumer appeal. As a result, we restricted our use of this information to a single director and the top five billed actors.

Data cleaning also posed difficulties due to missing or inconsistent values across multiple columns. Not all films listed five actors and one director, and in one extreme case, a movie included twelve directors. Throughout this process, we had to make decisions about which entries to retain or discard, knowing that these choices could directly influence model performance. For example, we chose to keep only English-language films because they represented the majority of usable entries, and including non-English films risked introducing large outliers shaped by cultural differences that our model was not designed to capture.

A final challenge involved quantifying the impact of actors and directors. Although we created actor and director score features, determining how to weight these values fairly—without leaking future information—proved difficult. We ultimately addressed this by examining the revenue of the five most recent films involving each actor or director and applying heavier weights to more recent works. This approach allowed us to incorporate past performance while respecting temporal order.

VI. Future Work

One of the biggest limitations of our work is the lack of non-English movies in the results. To obtain uniform outcomes that are not skewed by cultural biases, we chose to remove non-English movies. Additionally, the non-English portion of the TMDb dataset contains only around 500 entries, which barely represents the full range of global film production. For instance, using a comparable dataset focused solely on non-English films could yield interesting insights. To minimize cultural bias, each dataset would need to be uniform in language, especially for major international industries such as Bollywood in India, Nollywood in Nigeria, or large production studios in the United Kingdom.

Another potential direction for future work involves expanding the dataset. The TMDb database contains roughly 5,000 movies, whereas IMDb includes upwards of 700,000 titles. Some Kaggle datasets attempt to use IMDb data, but they are often removed due to copyright concerns. If a larger dataset can be obtained, the resulting models would better reflect the true population of films. This would likely increase accuracy by providing more information about which movies succeed or fail and the factors that contribute to those outcomes. Additional actor and director information would also strengthen the predictive value of those features.

A final direction for future work involves more extensive feature engineering. Advertising plays a key role in promoting a movie, yet our work does not incorporate marketing efforts beyond the budget itself, nor does it account for social media attention that develops outside traditional promotional spending. If these factors were represented properly, the model could better capture a movie's potential success by measuring the "buzz" it generates online, an influence that often encourages viewers to watch a film they may not have otherwise considered.

VII. Project Organization

This project was divided into two major components: data cleaning and preprocessing, and model building and evaluation. Across an approximately two-month period beginning after Fall Break, the majority of our time was devoted to preparing the dataset for modeling. We first aggregated the

columns from the two CSV files by performing an inner join on the movie ID field, ensuring that all information was contained within a single unified dataframe. We then replaced null values and removed features that were unlikely to influence model performance, including *homepage*, *overview*, *tagline*, *status*, *title_x*, *title_y*, and duplicate ID fields.

A substantial portion of the preprocessing effort involved parsing the nested JSON fields within the *cast* and *crew* columns. These fields were originally stored as stringified JSON objects and needed to be converted into usable Python structures. This step was especially time-consuming, and we were only satisfied with the processed results near the beginning of November. Afterward, we engineered actor and director score features by aggregating historical box office performance for each individual, enabling the model to incorporate past success as a predictive characteristic. We then restricted the dataset to films released before or during 2011 to support chronological training and avoid data leakage.

Model implementation began in mid-November, by which point the dataset was fully prepared. The model itself was completed within one week, requiring minimal refinement to achieve satisfactory performance. The remaining time was dedicated to generating performance metrics, producing graphs, and writing the final report. In total, the project spanned approximately two months, with the majority of the effort focused on cleaning and processing the dataset.

A. Team Contribution

- Anant Sahoo: Basic Processing; JSON Parsing; Report Writing
- Deep Patel: Feature Engineering; One-Hot Encoding; Report Writing
- Sulaiman Mohyuddin: Multi-Label Binarizer; Data Extraction; Report Writing
- William Douglass: Data Extraction; Feature Scaling; Figure Diagram Creation
- William Sessoms: Multi-Layer Perceptron Regressor; Hyperparameter Tuning

ACKNOWLEDGMENT

The authors would like to thank the Department of Electrical Engineering and Computer Science at the University of Tennessee, Knoxville, for providing a supportive research environment and facilitating valuable discussions that contributed to this work.

REFERENCES

[1] Kaggle, "TMDB Movie Metadata," Aug. 28, 2025. [Online]. Available: https://www.kaggle.com/datasets/tmdb/tmdb-movie-metadata/